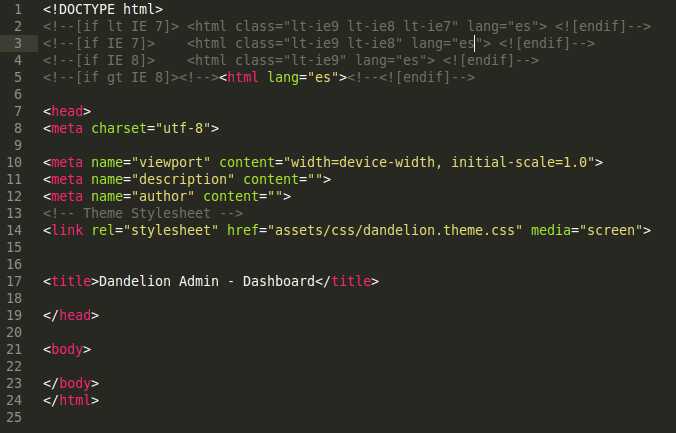
Manual de la plantilla alojada en el servidor nginx.

1. El documento HTML básico con el color de fondo en forma de cuadricula es el que se muestra en la imagen siguiente, y se encuentra en el template dashboard.html.

Nota: El código de la linea 2 a la linea 5 es para conseguir la compatibilidad con navegadores internet explorer de la versión 9 o anteriores.

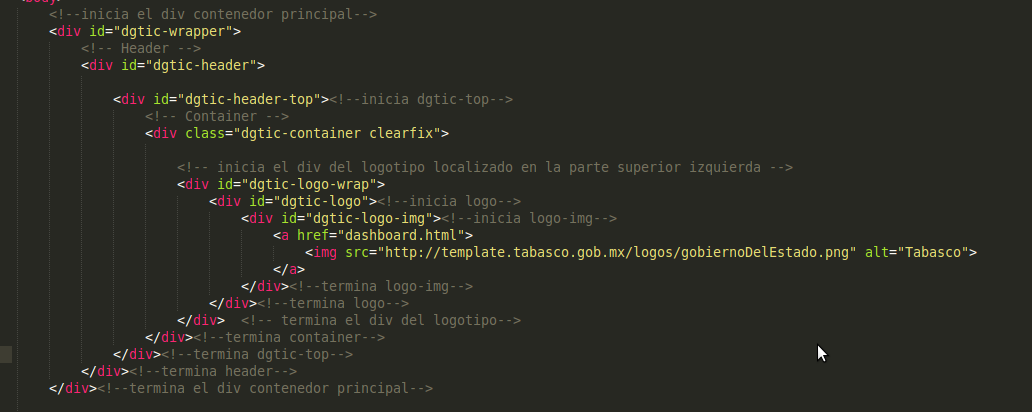
1.1 La linea 10 se declara para que la pagina pueda ser visualizada en dispositivos móviles.

1.2 Los estilos para que la pagina se vea el fondo gris y en cuadricula se encuentran en la hoja de estilos **dandelion.theme.css** especificada en la linea 14.

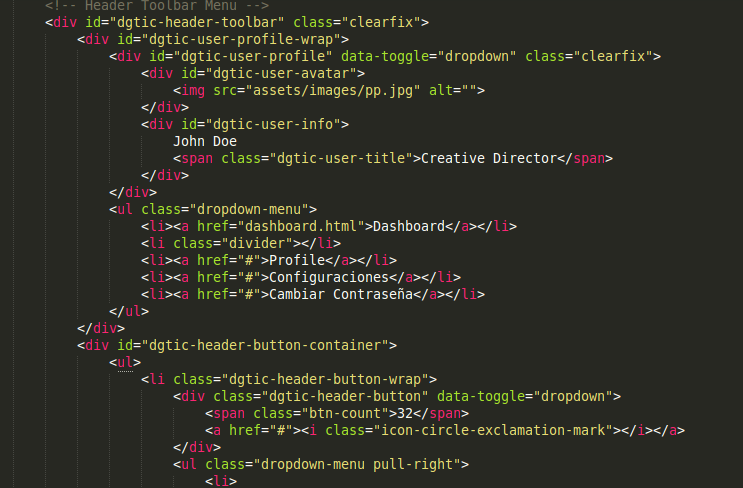
1. Los estilos css necesarios para que la pagina se vea como en la plantilla son los siguientes.

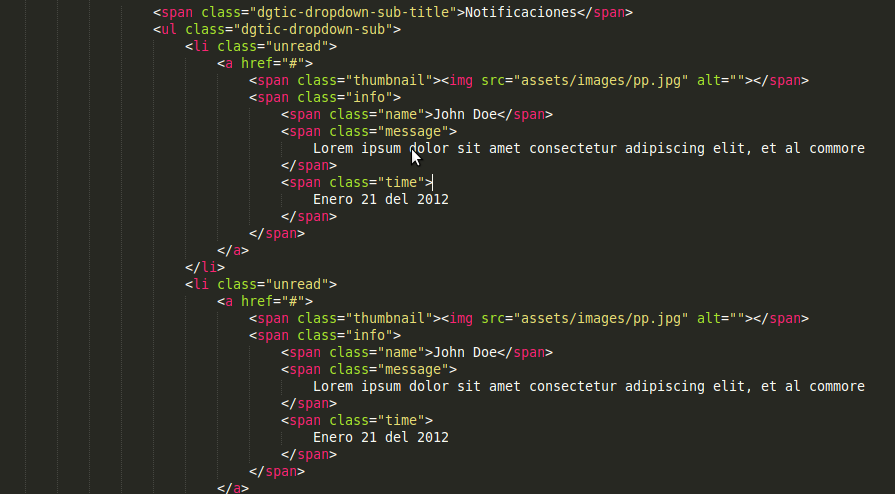
Nota: Se deben implementar tal y como se muestra en la imagen y deben de estar dentro del head.

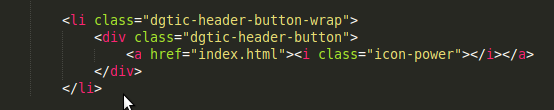
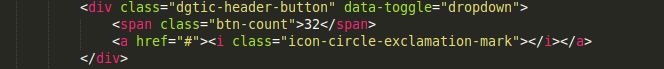
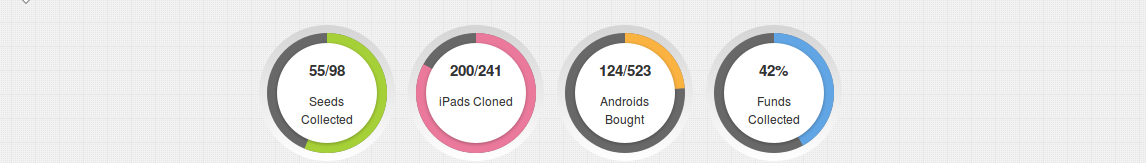
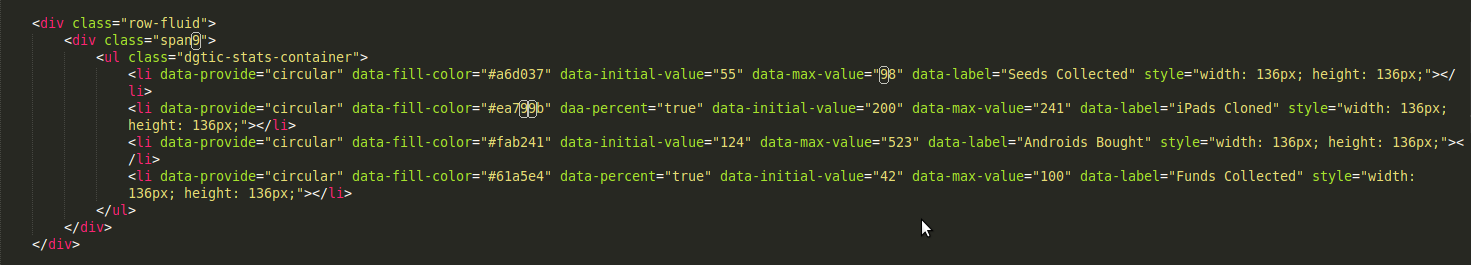
1. Los archivos js se deben nombrar después del cierre del ultimo **div** y antes del cierre del body para que funcione correctamente.
2. Para mostrar la cabecera principal con el logotipo se ingresa el código siguiente.

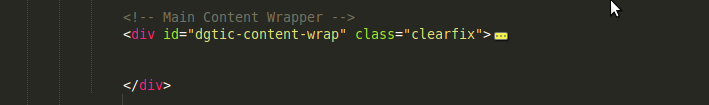
Nota: En la imagen anterior se muestra donde se habre y cierra los divs 

1. Para incluir los botones que se encuentran en la parte superior derecha de la plantilla se debe colocar el siguiente código que se muestra en la imagen siguiente.





1. Para el menú de cambio de contraseña que se activa al hacer clic en la flecha que esta alado del nombre del usuario se debe de ingresar el siguiente código que se muestra en la imagen.
2. El código que crea el botón de salir y los que están a lado se muestra en la siguiente imagen.
3. El paso 7 aplica para los botones de mensajes y notificaciones solo se cambia la clase que se le aplica a cada elemento. Ejemplo para el botón de notificación:
4. Para la creación de gráficas circulares como las que se muestran en l imagen siguiente.
5. Es necesario ingresar el codigo siguiente para que funcione.
6. El valor del **data-initial-value** puede ser cargado desde una consulta a la base de datos y sirve para indicar el porcentaje actual en la gráfica, el data-max-value. Indica el valor máximo en porcentajes y también puede ser llenado con información de la base de datos. El cálculo lo realiza en automático sin realizar otra operación.
7. Para los bloques o divisiones de la pagina existen varias medidas y se puede implementar seleccionando uno de los diferentes tipos de **span**, ya que van del **span1** al **span12.**
8. Para los bloques divisionales es el siguiente :



1. Existen 4 tipos de botones en la plantilla los de color rojo está indicado para cancelar o cosas similares el código completo para mostrar el botón es el siguiente.

**<input type="submit" value="Submit" class="btn btn-danger">**

El botón verdeesta indicado para guardar un formulario el código para crear el botón es el siguiente.

**<input type="submit" value="Submit" class="btn btn-success">**

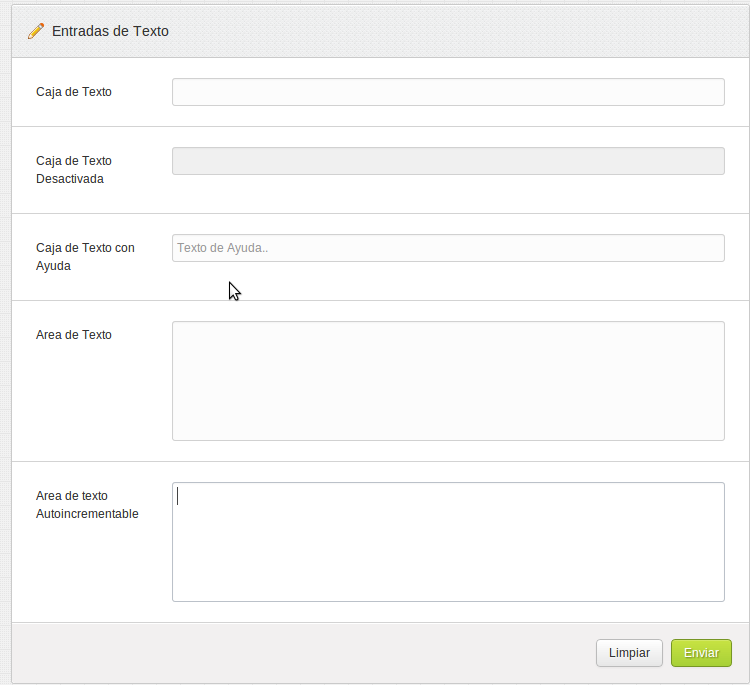
el botón de color azul esta indicado para actualizar un formulario o cosas similares el código que lo forma es el siguiente.

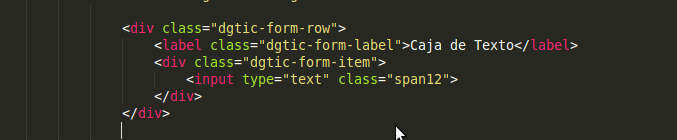
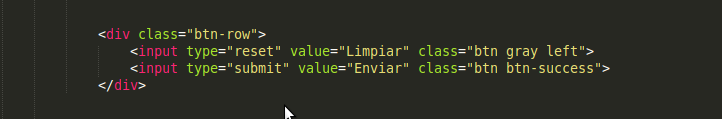
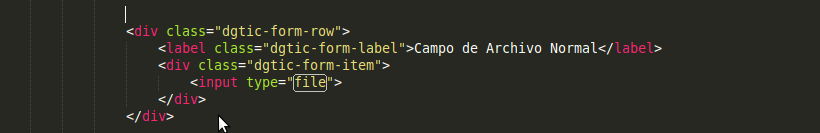
**<input type="submit" value="Submit" class="btn btn-primary">**

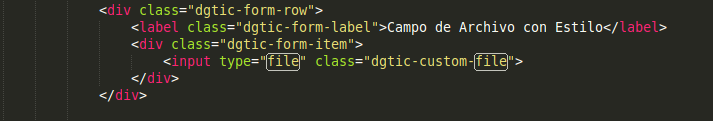
Existe un botón de color gris que esta indicado para limpiar los formularios, el código que lo forma es el siguiente.

**<input type="reset" value="Reset" class="btn gray left">**

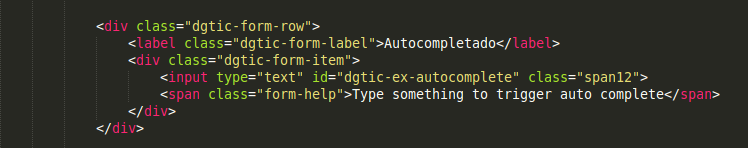
1. Para que los formularios se muestren como en la imagen siguiente.

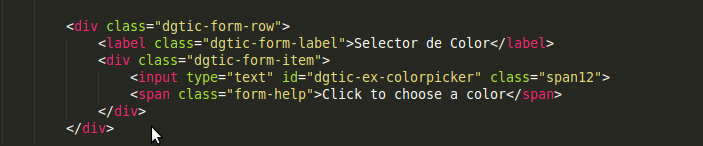


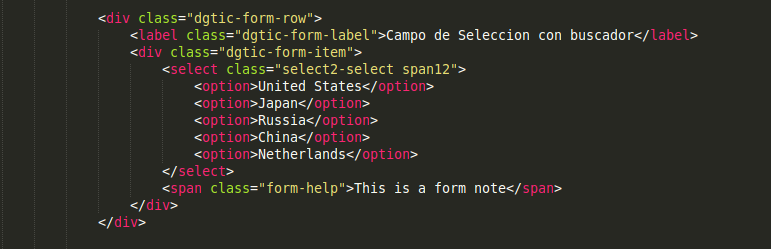
1. Se deberá utilizar el siguiente código para cada Label y para cada input.
2. El código que crea los botones es el siguiente.
3. Para los campos de tipo file sin estilos se debe utilizar el siguiente código.
4. Para los campos de tipo file con estilos se debe de utilizar el siguiente.



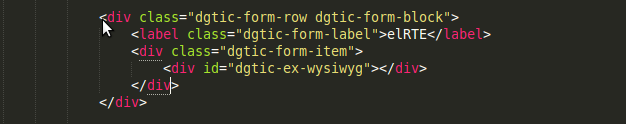
1. Para los campos de texto con auto completado se debe de implementar de la siguiente manera.

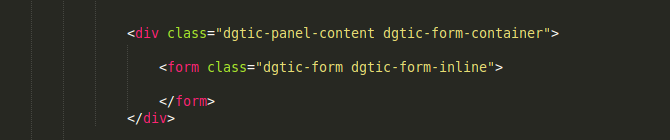
Nota: el auto-completado funciona mediante una función jquery que se activa al ingresar txto en el campo y es identificado por el id “**dgtic-ex-autocomplete”**.

1. Para mostrar un campo donde se permita seleccionar un color de una paleta de colores se debe de ingresar el siguiente código.
2. El selector de color se implementa ejecutando una libreria jquery que se activa al hacer referencia al id=**”dgtic-ex-colorpicker”.**
3. Para implementar los select con un buscador el código es el siguiente.

Nota: Los option pueden ser llenados con información de una base de datos.

1. Si se requiere implementar un área de texto enriquecido, se puede utilizar el código siguiente.

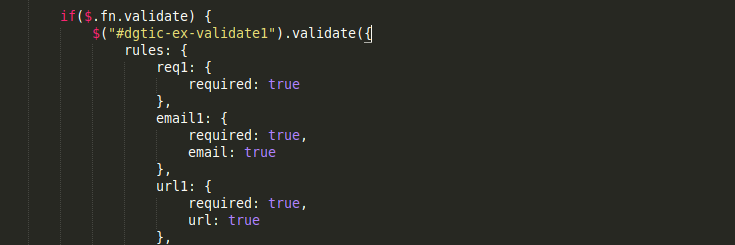


1. Los códigos de las imágenes 15 a la 24 deberán estar dentro del siguiente código para que funcione correctamente.
2. Para los demás elementos del formulario su implementación es similar al de las imágenes anteriores solo se debe de declarar el tipo de elemento en cuestión.

**Validación de Formularios**

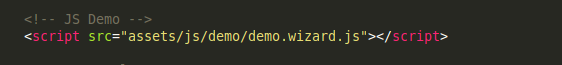
1. Cuando se requiera validar un formulario o varios formularios se debe de implementar la siguiente librería JavaScript.

Nota: Para que la validación funcione se tiene que colocar un **id** al formulario para diferenciar si existen mas de un formulario en el mismo documento html, ya que la función javascript que ejecuta la validación reconoce a cada formulario por su **id.**

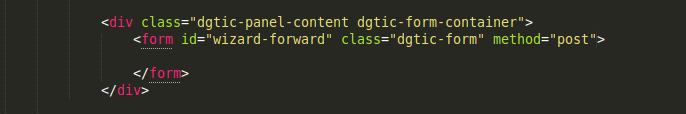
1. Para adaptar la función JavaScript a cada formulario se puede modificar el archivo demo.validacion.js, ahí se indica los tipos de datos, los elementos del formulario a validar,etc. **Ejemplo:**
2. En el archivo demo.validacion.js se pueden agregar las validaciones que se necesiten.

**Pasos para la implementación del wizard.**

1. De debe se agregar la librería JavaScript

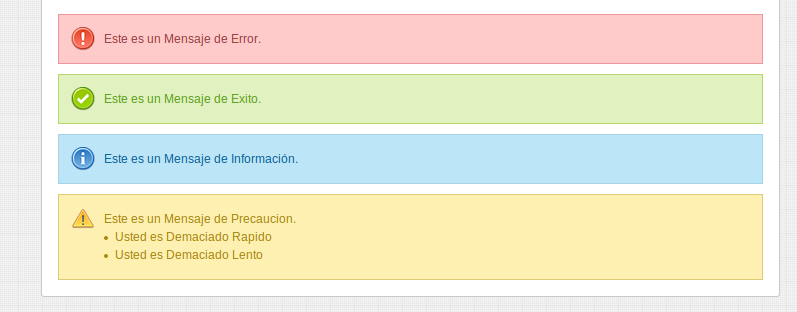
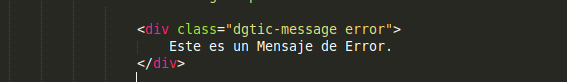


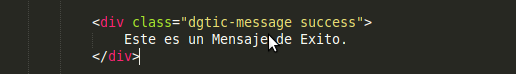
1. Para formar el wizard se debe agregar el siguiente código.

1.- El formulario debe llevar un id=**”wizard-forward”,**por medio del id la librería lo convierte en un widzard.

2.- Deberá tener una clase  **“dgtic-form”** par que se le apliquen los estilos CSS.

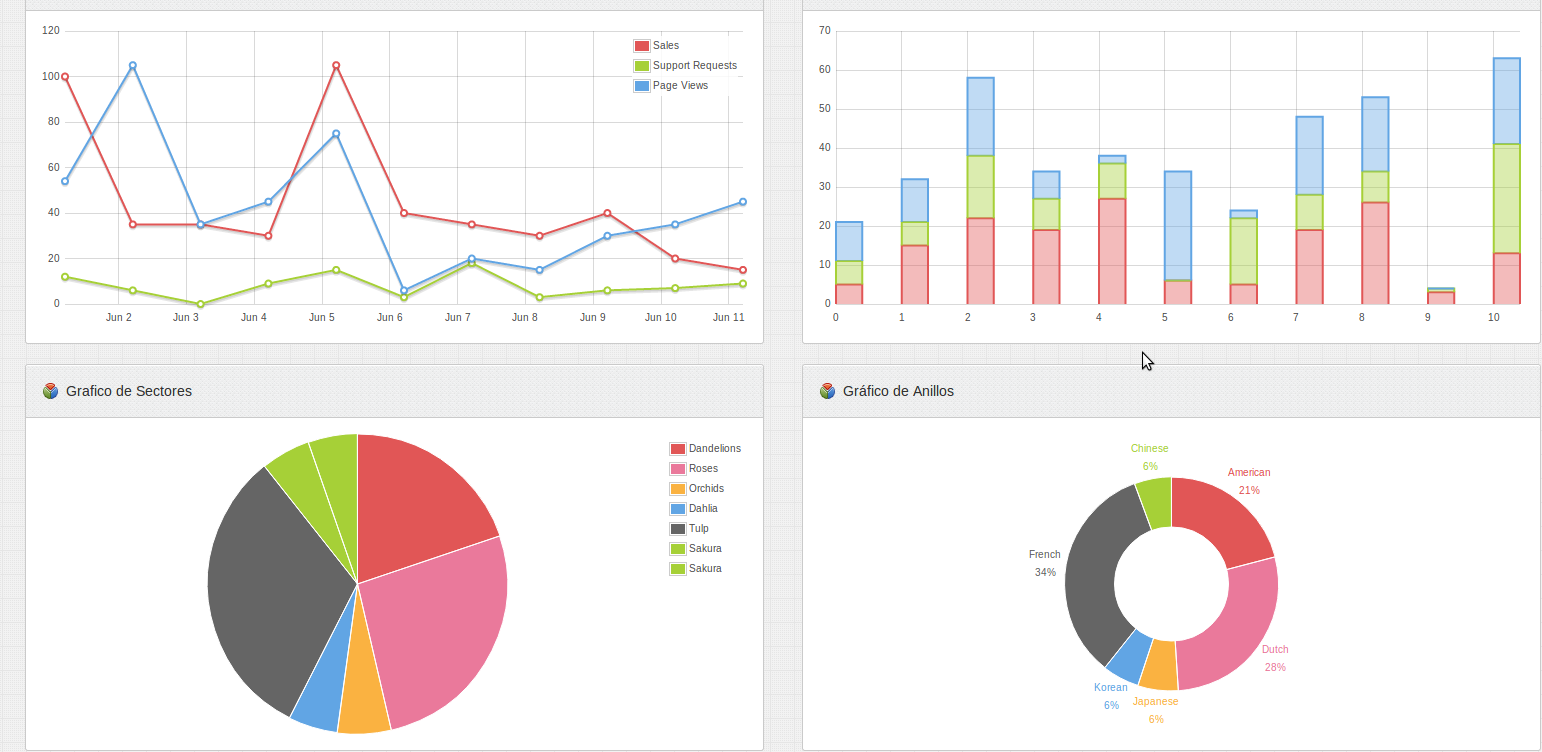
3.- Los botones **prev** y **next** no son necesarios agregarlos por que la librería los agrega.

1. Existen 4 tipos de mensajes diseñados, para mostrar un color según sea el evento. Ejemplo:
2. Para el mensaje de error el código fuente es el siguiente. 
3. El mensaje de éxito que en la imagen anterior se muestra de color verde se implementa con el código siguiente.



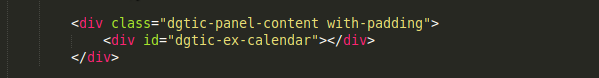
Para los otros dos mensajes es similar a los anteriores solo deberá cambiar el nombre de la clase

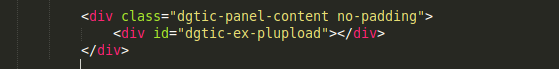
por los siguientes. Para los mensajes de Información la clase debe quedar **class=”dgtic-message info”, para los mensajes de precaución se deberá aplicar class=”dgtic-message warning”.**

1. Las gráficas que se muestran en las plantillas en el apartado estadísticas.
2. **Ejemplo:**
3. Son creadas con la librería Canvas con **jquery.flot.js**  para su implementación se debe manejar jquery.
4. **Si se desea mostrar el calendario que se muestra ene la plantilla se deve agregar la libreria siguiente.**

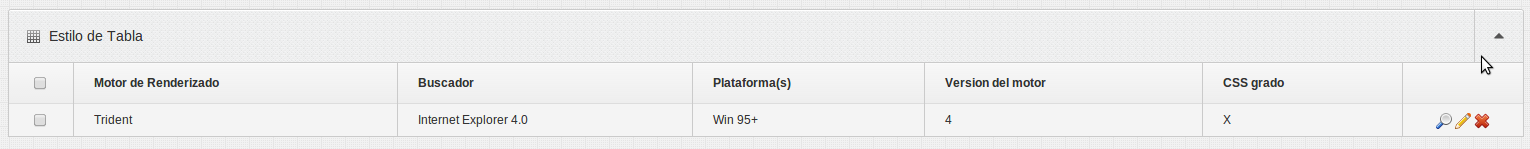


1. Y luego agregar un div en el documento html como se muestra a continuación.

****

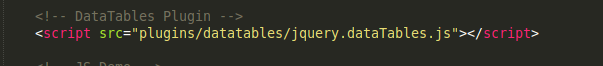
1. **Nota:** El código anterior puede estar dentro de un panel o un panel collapsible, la libreria jquery antes mencionada es la que forma el calendario.
2. El código anterior sirve para implementar un formulario para subir archivos al servidor, solo se crea el div y el plugin de jquery se encarga de armar el siguiente formulario.



1. Para implementar las tablas que se encuentran dentro de un bloque collapsible como se muestra en la siguiente imagen.
2. El código necesario es el siguiente.
3. Si se desea implementar las tablas con ciertos datos por defecto y colocarle un buscador se realiza de la misma forma que en el paso 43 cambiando el id y la clase a la tabla por el siguiente.

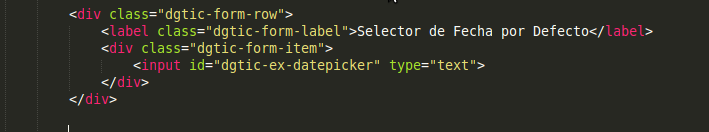
**<table id="dgtic-ex-datatable-default" class="dgtic-table">**

1. Para que las tablas queden tal y como se muestran en la plantilla se debe importar la librería siguiente.



1. Los calendarios se implementan importando la librería siguiente.

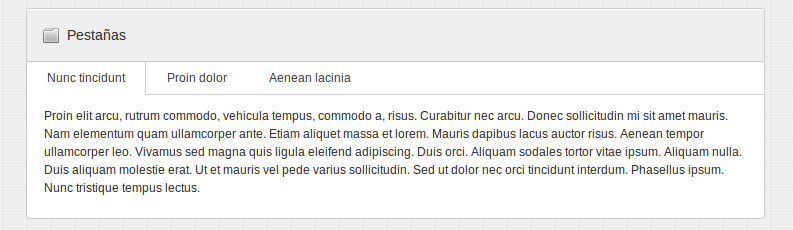
**<script src="assets/jui/jquery.ui.timepicker.min.js"></script>**

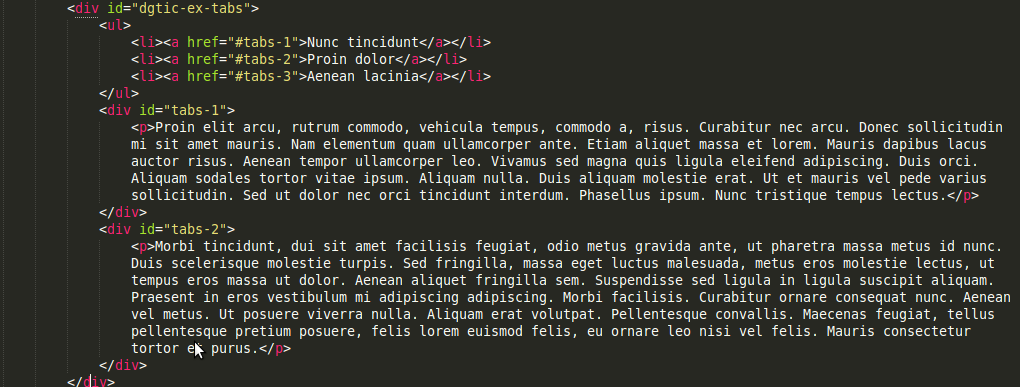
****

1. En la imagen anterior se muestra la forma en que se deb de implementar este código debe estar dentro del código que se muestra en el paso 26.
2. Para la implementación del elemento hora y fecha es similar al de calendarios solo se cambia el id del input por el siguiente: **id=**”**dgtic-ex-datetimepicker**”.
3. Para la implementación del calendario con el numero de la semana solo se cambia el id del input por el siguiente: **id=”dgtic-ex-datepicker-week”**.
4. Para la implementación del calendario con más de un mes se coloca el siguiente id al input.

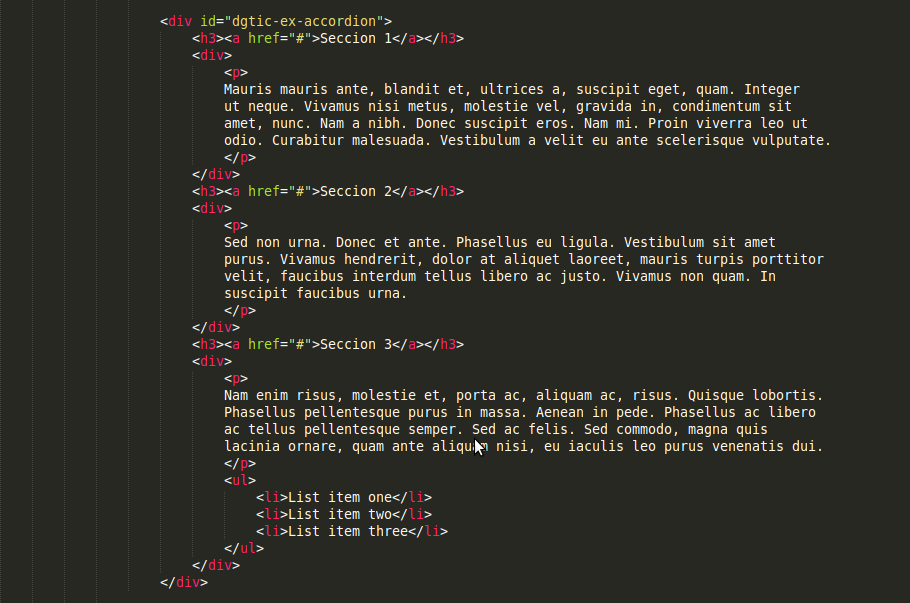
**id="dgtic-ex-datepicker-months”**.

**Implementación de Pestañas**

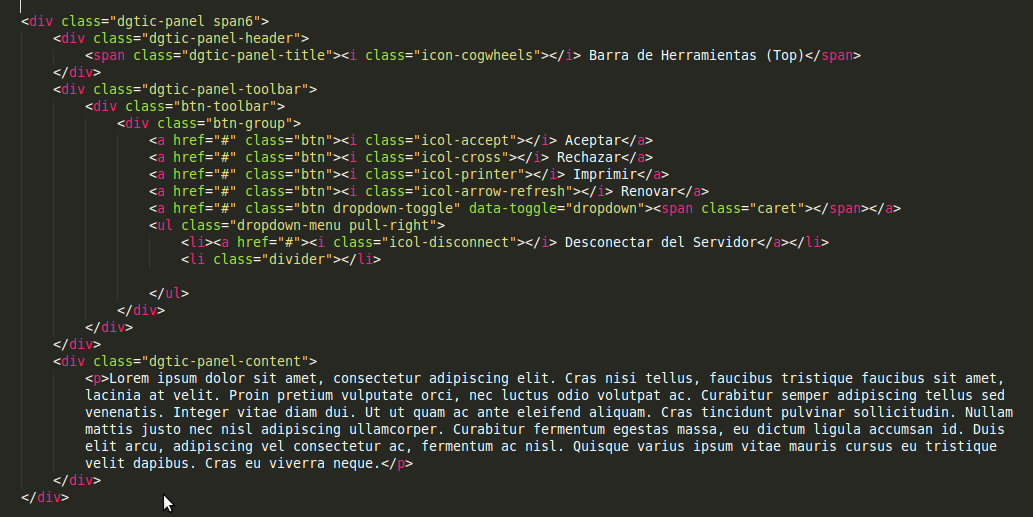
1. Para implementar la navegación por pestañas en un formulario como el que se muestra en la siguiente imagen.
2. El código que forma la navegación por Pestañas es el Siguiente.



1. Para agregar mas pestañas se tiene que agregar un **<li><a href=”pestaña”>pestaña</a></li >** por cada pestaña que se requiere que aparezca y por cada **<li>** agregado se tiene que agregar un **<div>** que debe contener en el id el nombre del **href** creado en los **<li>**.
2. Para mostrar la navegación por acordiones se implementa el siguiente Código dentro de un panel de los mostrados anteriormente.



1. Implementación de pestañas con barra de herramientas.
2. Para implementar las pestañas con barra de herramientas dentro de un panel ya sea collapsible o no se debe ingresar el siguiente código dentro del código del panel.

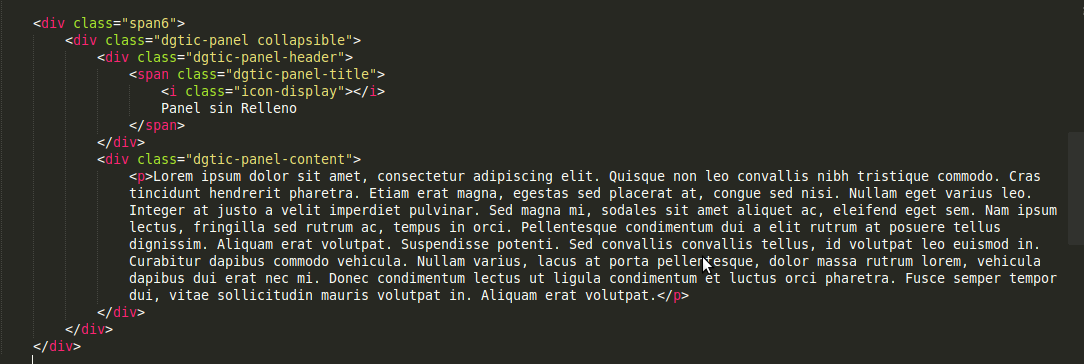
****

1. Las barras de progreso se implementan mediante la librería jquery por lo que para cargar datos dinámicos traídos de una consulta se tiene que ingresar directamente en el código jquery.
2. Para implementar las herramientas de ayuda en los botones o otros elementos(tooltip). Para ello el código es el siguiente.



1. Implementación de Paneles.

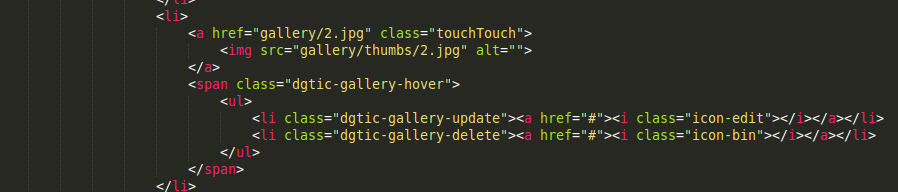
1. Para los paneles collapsible se implementan con el Código siguiente.



1. Para los paneles no collapsible se implementa con el siguiente código.
2. Estos paneles se activan con jquery y deben los divs deben de estar dentro de los div principal.

Para los tipos de letras se importa la siguiente librería.

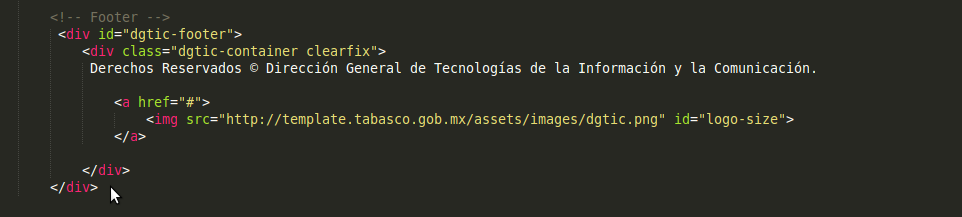
1. Para implementar las galerías de imágenes se importa la siguiente librería.
2. El codigo que debe ir en el documento html para formar la galería es el Siguiente.
3. Dentro de las lineas de código anterior se deberá implementar una lista con las imagenes a mostrar. Ejemplo.



1. Para los iconos que que se muestran en los enlace o en el apartado iconos de la plantilla se implementan con el siguiente código.



1. Donde el elemento que muestra la imagen es <i> y para que muestre la imagen que se necesite se le aplica el nombre a la clase como se muestra en la imagen anterior.
2. El pie de pagina se configura con el siguiente código el cual deberá estar antes del cierre del ultimo div del documento html como se muestra en la imagen siguiente.

Nota: La configuración de la ruta de la imagen depende de la configuración de **settings.py** del proyecto en cuestión.